

Northumbria Research Link

Citation: Eliot, Neil, Kendall, David and Brockway, Michael (2018) A New Metric for the Analysis of Swarms using Potential Fields. IEEE Access, 6. pp. 63258-63267. ISSN 2169-3536

Published by: IEEE

URL: <https://doi.org/10.1109/ACCESS.2018.2877421>
<<https://doi.org/10.1109/ACCESS.2018.2877421>>

This version was downloaded from Northumbria Research Link:
<http://nrl.northumbria.ac.uk/id/eprint/36414/>

Northumbria University has developed Northumbria Research Link (NRL) to enable users to access the University's research output. Copyright © and moral rights for items on NRL are retained by the individual author(s) and/or other copyright owners. Single copies of full items can be reproduced, displayed or performed, and given to third parties in any format or medium for personal research or study, educational, or not-for-profit purposes without prior permission or charge, provided the authors, title and full bibliographic details are given, as well as a hyperlink and/or URL to the original metadata page. The content must not be changed in any way. Full items must not be sold commercially in any format or medium without formal permission of the copyright holder. The full policy is available online: <http://nrl.northumbria.ac.uk/policies.html>

This document may differ from the final, published version of the research and has been made available online in accordance with publisher policies. To read and/or cite from the published version of the research, please visit the publisher's website (a subscription may be required.)



**Northumbria
University**
NEWCASTLE



UniversityLibrary

Received October 5, 2018, accepted October 16, 2018, date of publication October 22, 2018, date of current version November 14, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2877421

A New Metric for the Analysis of Swarms Using Potential Fields

NEIL ELIOT¹, DAVID KENDALL¹, AND MICHAEL BROCKWAY¹

Department of Computing and Information Sciences, Northumbria University, Newcastle upon Tyne NE1 8ST, U.K.

Corresponding author: Neil Eliot (neil.eliot@northumbria.ac.uk)

ABSTRACT There are many metrics defined for the analysis of swarm coordination algorithms. These metrics are usually based upon the distances between agents, the distance between agents and a fixed point, or the resultant vectors that *potential field* effects produce. This paper examines a distance-based metric that measures a swarm's overall structure using inter-agent distances. More importantly, it introduces a new metric that identifies a swarm's state based upon the resultant magnitude of the vectors produced by the agent interactions that create the agent distribution within the swarm's structure. The algorithms used to implement the swarming feature are based upon cohesion and repulsion vectors between an agent and its neighbors. In comparing and contrasting the two metrics, we find that the *cohesion/repulsion* metric offers a number of advantages over the distance metric. In particular, the *cohesion/repulsion* metric allows the identification of the essential characteristic of a swarm as "expanding," "stable," or "contracting." These states cannot be identified using a distance-based metric. Practical swarming applications where the new metric can be applied advantageously include area-filling and reconnaissance.

INDEX TERMS Swarming, swarm dynamics, swarm metric.

I. INTRODUCTION

Swarming in the animal kingdom of ants, bees, fish and birds, for instance, has long been studied by scientists. From these studies, mathematical models and algorithms have evolved, such as those produced by Reynolds [24]. The models and algorithms have in turn captured the interest of computing scientists, who are interested in applying them to large groups of autonomous mobile *agents* ('robots'). The cooperative coordination of these agents can take many forms, such as following a set path [17], existing in a static space [9], [11], [12] or foraging as a colony [15], [16]. One of the attributes of swarms that has captured the interest of scientists is that the models and algorithms used to coordinate them are generally sets of simple rules. These simple rules cause the agents to appear to work cooperatively. Often, the rules can also generate movement that does not contribute to the goal of the swarm. This movement is a direct result of the agents attempting to move towards an optimum position within the structure. This internal movement can be identified by analyzing the changes in the inter-agent interactions. Measuring this unnecessary movement allows the effectiveness of a swarming algorithm to be evaluated. If the internal movement can be reduced, then the stability of the swarm is improved and more of the swarm's energy

is focused on the goal. Reducing the internal movement also provides a more stable platform for the deployment of sensor arrays. Both metrics, presented in this paper, allow this evaluation to be carried out when all the sensor ranges are the same. However, the distance metric is not able to identify the internal movement accurately when the sensor ranges of each agent vary, producing irregular inter-agent distances, even though the interactions at the vector level indicate that the agents are in fact stable and optimum. This paper discusses the two metrics and how they differ in their approach to identifying these changes. *Distance-based* metrics use variations in the inter-agent spaces, as discussed by many research teams including Navarro and Matía [22], Gazi and Passino [11], [12], Barnes *et al.* [6], and Schneider and Wildermuth [26]. The new metric uses the resultant magnitudes of the agents' *inter-agent vectors* that are defined by interactions of an agent with its neighbors. This allows the "degree" of a relationship to be evaluated globally and therefore allows a swarm to be identified as "expanding", "stable" or "contracting".

Many coordination algorithms exist to control the movement of agents within a swarm. The primary goal of these algorithms is to maintain a stable swarm structure. A stable structure is one where the agents in the swarm "act" as a

single entity. The main focus of this paper is on metrics that can be applied to any coordination algorithms that use *potential fields* to produce a swarming effect. Most swarming algorithms define a set of equations that calculate the repulsion and cohesion effects. These equations are the basis of the new metric.

To facilitate the analysis of the new metric, a basic swarm model is introduced and then two metrics are applied to this model to present a comparison of the information provided by a *distance-based* and the new *cohesion/repulsion-based* metric. The analysis includes the identification of a swarm's "state". The metric is then applied to an area-filling scenario to highlight a possible use of the new metric.

The remainder of this paper is structured as follows: Section II discusses different metrics that are currently used to analyse swarms; Section III introduces a model of swarm behavior, using potential fields, that can be analyzed and evaluated using the two metrics; Section IV shows examples of the typical dynamic behavior of swarms under the potential field model; Section V discusses a distance-based metric for swarm analysis; Section VI introduces the new *cohesion/repulsion-based* metric, showing how it is calculated and applied; Section VII compares the two metrics using an area-filling scenario; Section VIII discusses the results of experiments, and, finally, Section IX discusses additional work that needs to be carried out to exploit the additional information that the new metric provides.

II. RELATED WORK

Many metrics exist for the analysis of swarms. These metrics can be categorized as *centroidal-based*, *agent/inter-agent distance-based* and *magnitude-based*. The common denominator of *centroidal* and *agent/inter-agent distance* metrics is that they *measure* the agent distribution within a swarm based upon distances. *Magnitude-based* metrics analyse a swarm based upon inter-agent interaction.

Centroidal analysis involves identifying the centroid of a swarm, also referred to as the *center of mass*, as shown in Figure 1 as a red cross and defined in Equation 1.

$$\bar{r} = \frac{1}{|S|} \sum_{b \in S} r_b \quad (1)$$

Here, \bar{r} denotes the position vector of the swarm's *centroid*; r_b is the position of each agent, b , and S is the set of agents that make up the swarm.

Navarro and Matía [22] use the centroid to produce a "Mean Distance from the Boundary to the Center of the Group" metric. This metric takes the distance of each agent from the centroid and calculates the mean and deviation to produce their measure. Gazi and Passino [14] use this technique when analyzing the stability of a swarm based on agent distribution. The *centroid* is also used as a reference point to determine the path of a swarm. Gazi and Passino [12] use the centroid when analyzing the stability of a swarm which is designed to maintain its position, by analyzing the changes in the centroid's position over time. Gazi and Passino [11] also

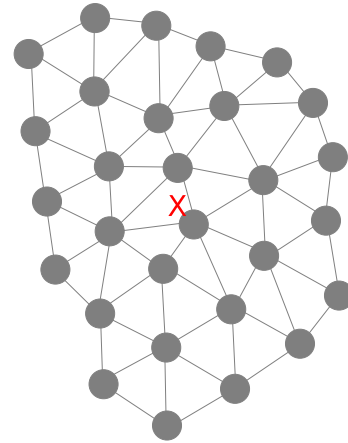


FIGURE 1. Swarm Centre of Mass.

use a centroid-based analysis to identify directional changes in agent movement by creating a "map" of the potential fields of a swarm in its entirety and overlaying that to the "centre" of the swarm. Barnes *et al.* [5], [6] use the centroid to analyse the effects of using bivariate normal functions in their work on using potential fields.

Inter-agent distance analysis is a basic mechanism for analyzing a swarm in terms of separation of all of the agents with respect to each agent's visible neighbors. This technique is the most popular approach to analyzing a swarm's structure. Gazi and Passino [12], [14] use a distance-based analysis to identify optimum distribution of agents based upon different cohesion and repulsion applications. Wu *et al.* [27] discuss using distance-based metrics to evaluate emergent behaviors in swarms based on a Viscsek model. These include calculating the average number of neighbors for each agent, grouping agents based on numbers of neighbors (to identify clusters), and calculating the average distance between agents [27]. Schneider and Wildermuth [26] use a slightly different approach to measuring performance. They identify the difference in the paths of robots in a simulated versus a physical implementation and calculate a distance variance between them.

Magnitude-based analysis uses the *potential fields* to calculate the "degree" of interaction between agents. Bennet and McInnes [8] use the resultant relationship between agents to identify equilibrium states when creating structures such as rings and controlling the direction of agents to create specific transformations. Wu *et al.* [27] introduce the concept of a *rejection potential* in an attempt to generate a "kinetic energy" score for a swarm.

This paper builds upon the work relating to the *magnitude-based* metric analysis, as discussed by Bennet and McInnes, and Wu *et al.*, by expanding the analysis to the entire swarm and deriving a scalar value that can be used in the analysis of swarm behavior.

III. SWARM MODELLING WITH POTENTIAL FIELDS

Currently, much swarm research uses field effects as the method of modeling inter-agent interactions [3], [5], [6], [8],

[10]–[14], [21]. The models usually use two field effects to implement the swarming characteristic. These effects are *cohesion*, to draw agents closer, and *repulsion* to prevent agents colliding. *Potential fields* are ranges around an agent that determine the effect other agents have upon its movement (Fig. 2). It is usual for the cohesion field to have a radius C_b which is larger than the repulsion radius R_b . When an agent b' moves into the *cohesion field* of an agent b , then b' is said to be a neighbor of b , and b is subject to cohesion. When an agent b' moves into the *repulsion field* of b , then b is subject to repulsion. When the repulsion magnitude exceeds the cohesion magnitude, the agent b will have a tendency to move away from b' , i.e. to be repelled. Obstacles can be introduced into the model. When an agent b moves too close to an obstacle, i.e. within the obstacle repulsion range O_b , it has a *repulsion vector* applied, giving it a tendency to move away from the obstacle.

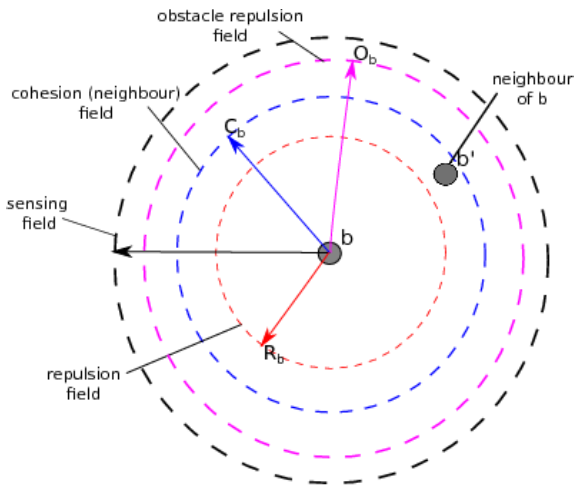


FIGURE 2. Agent field effects.

Figure 3 shows the cohesion and repulsion vector contributions of neighbor b' to $v_c(b)$ and $v_r(b)$, as given in Equations 2 and 3. Notice that the vectors are along the line of separation bb' .

Equation 2 identifies the resultant cohesion effect of all the neighbors of b . Here, $nbr(b)$ defines the set of all the agents

in the swarm that are a neighbor of b ; a neighbor is any agent that falls within the cohesion field.

$$v_c(b) = \frac{1}{|nbr(b)|} \left(\sum_{b' \in nbr(b)} bb' \right) \quad (2)$$

Equation 3 identifies the resultant repulsion effect of all the neighbors of b , where $rep(b)$ returns the set of all agents that are within the repulsion field of b .

$$v_r(b) = \frac{1}{|rep(b)|} \left(\sum_{b' \in rep(b)} \left(1 - \frac{|bb'|}{R_b} \right) bb' \right) \quad (3)$$

Using the cohesion and repulsion vectors generated by the relationship of b to its neighbors, a resultant vector can be calculated that summarizes the effects to which the agent is being subjected, and provides the direction in which the agent will move, based on this relationship. This is the *inter-agent vector*. Equation 4 identifies the *inter-agent vector* for agent b with respect to its neighbors.

$$v(b) = v_c(b) + v_r(b) \quad (4)$$

If a *weighted* model is used then Equation 4 is replaced by Equation 5, where k_c and k_r are weightings to change the effect of each vector, as exemplified in Table 1.

$$v(b) = k_c v_c(b) + k_r v_r(b) \quad (5)$$

TABLE 1. Swarm model parameters.

Weight Component	Setting	Description
Sample Rate	100	ms - Unit sampling interval
k_c	5	weight adjuster for cohesion bias
k_r	15	weight adjuster for repulsion bias
k_d	0	weight adjuster for directional bias 0 for static baseline 100 from directional
Repulsion Boundary	70	units
Neighbour Distance	80	units
Speed	20	units/s

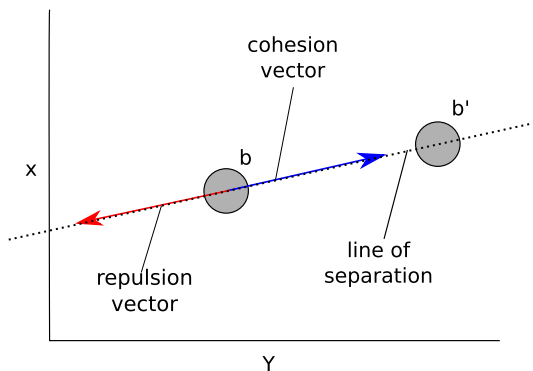


FIGURE 3. Vectors on line of separation.

When the two vectors (cohesion and repulsion) have magnitudes that are equal and opposite they produce a null vector, indicating that two agents are optimally spaced for a given set of conditions. Although the agents are at an optimum position when resultants are zero it does not mean the swarm is optimally distributed. If a swarm is in a confined space it is possible for an optimum position to be created where the vector magnitude is affected by a compression effect. This phenomenon is used in the identification of the emergent behavior of *area filling*.

If we consider the equilibrium state, the resultant $v(b) = (0, 0)$. A null vector cannot be normalized to produce a directional vector ($\hat{v} = \frac{v}{|v|}$ if $v \neq 0$; 0 if $v = 0$) The effect is that

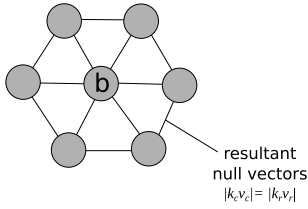


FIGURE 4. Equilibrium with null vectors.

the agent will remain stationary. If all agent pairs are in this condition the swarm will stop moving (Fig. 4).

Due to the movement of agents being based only on neighbor interactions, this situation is very rare. The residual motion that persists in a swarm is the background ‘noise’ or ‘jitter’ that an algorithm creates.

A swarm can also have a *goal*. This is modelled simply by the addition of a *directional vector*, as shown in Equation 6.

$$v(b) = k_c v_c(b) + k_r v_r(b) + k_d v_d(b) \quad (6)$$

Here, $k_d v_d(b)$ is the calculated directional vector to be applied to an agent.

Goal-based swarms tend to prevent all agents simultaneously producing null vectors, as suggested by Fig. 5 and 6.

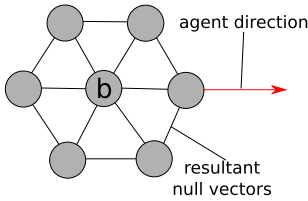


FIGURE 5. Equilibrium with directional vector (t).

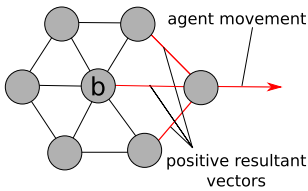


FIGURE 6. Equilibrium with directional vector (t + 1).

IV. SWARM DYNAMICS

Each agent within a swarm calculates its *movement-direction vector* based on its *inter-agent* and *directional* vectors. The *movement vector* (b_{pos}) is calculated using the unit *movement-direction vector* (Equation 6), multiplied by the time elapsed (t) in the system, and the speed characteristic of the agent (s_b).

This process is carried out for every agent in the swarm to generate the swarm’s next position.

$$b_{pos} = s_b t \widehat{v(b)} \quad (7)$$

The change in the location of agent b , over time interval t , is shown in Equation 7, where s_b is the speed of agent b . Note that \widehat{v} is the equivalent of $\hat{v} = \frac{v}{|v|}$, the normalized vector.

The repulsion and cohesion vectors are generated for an agent through the interaction of its field effects (Fig. 2). There are a limited number of interactions that can occur. These are illustrated in Fig. 7, 8, 9 and 10.

TABLE 2. Data extract ($|k_r v_r| = 0$).

Log	Id	N.Id	Distance	Cohesion	Repulsion
0	1	3	70.50359	352.51799	0
0	1	100	71.78005	358.90027	0
0	1	151	78.33995	391.69979	0
0	2	99	72.04066	360.20334	0

Following repeated experiments in the initial deployment of a swarm, it was found that, using the same parameters for a swarm’s environment and cohesion and repulsion ranges, always produced very similar behavior. The results shown in this section are an example set typical of the results obtained. The data extracts shown in Tables 2, 3, 4 and 5 are the simulation results that are produced by the parameters listed in Table 1. The simulation consists of 200 agents over a 20 second period. The cohesion of an agent pair is shown as $k_c v_c$ and the repulsion as $k_r v_r$.

Figure 7 shows two agents within each other’s cohesion fields but sufficiently distant to be outside of the repulsion fields. In this case $|k_c v_c| > 0$ and $|k_r v_r| = 0$: the result is that the agent’s resultant magnitude $v(b)$ causes the agent to move towards its neighbor b' . Likewise the neighbor’s resultant magnitude $v(b')$ will cause it to move towards b . Table 2 shows the repulsion magnitude with a value of 0. The only influence on the agent pair are cohesive vectors.

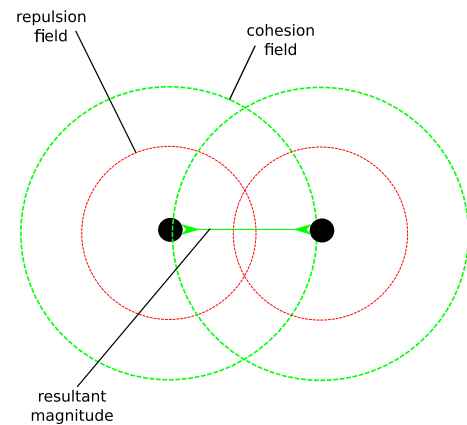


FIGURE 7. Internal movement cohesion (no repulsion).

Figure 8 shows two agents close together with repulsion dominating cohesion: $|k_c v_c| < |k_r v_r|$. The resultant vector directs the agents away from each other.

Table 3 shows the repulsion magnitude with a value greater than the cohesion magnitude.

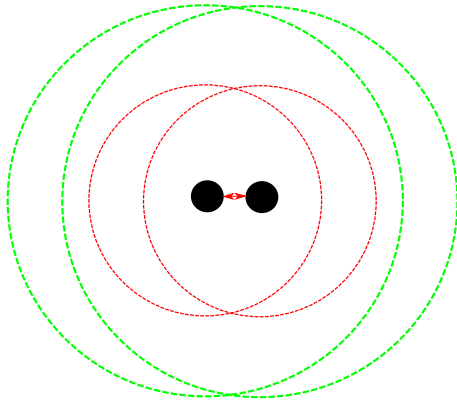


FIGURE 8. Internal movement repulsion.

TABLE 3. Data extract ($|k_c v_c| < |k_r v_r|$).

Log	Id	N.Id	Distance	Cohesion	Repulsion
0	1	2	28.32522	141.62612	1544.86014
0	1	6	41.48517	207.42586	721.71736
0	1	7	35.26412	176.32064	1034.27101
0	1	8	43.54503	217.72518	637.90759

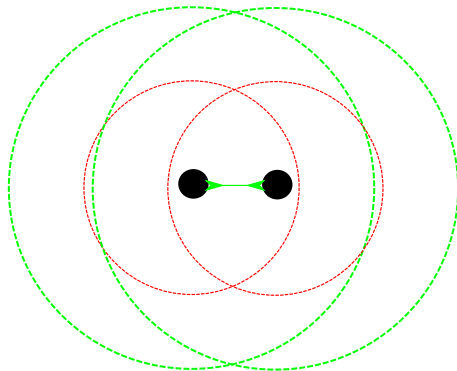


FIGURE 9. Internal movement cohesion and repulsion.

Figure 9 shows two agents close together but with cohesion vector magnitudes greater than the repulsion magnitudes $|k_c v_c| > |k_r v_r|$. The resultant vector draws the agents together. The magnitude of the resultant cohesion vector reduces due to the cancelling effect of the repulsion vector. Table 4 shows a data extract with the cohesion magnitude greater than repulsion.

TABLE 4. Data extract ($|k_c v_c| > |k_r v_r|$).

Log	Id	N.Id	Distance	Cohesion	Repulsion
0	1	5	64.17214	320.86072	95.35676
0	1	9	63.88049	319.40248	100.58590
0	1	95	65.61522	328.07613	70.16681
0	1	152	63.10700	315.5350	114.68844

Figure 10 shows two agents close together with $|k_c v_c| = |k_r v_r|$. The resultant vector is a *null vector* and the agents have no overall influence upon each other, since the

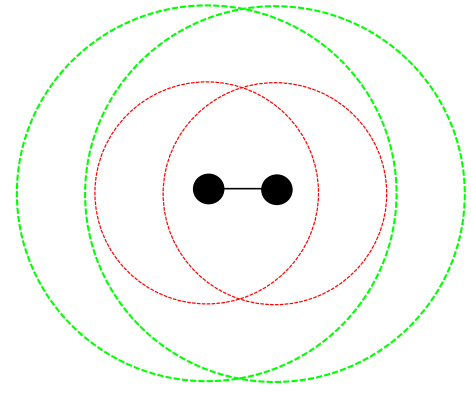


FIGURE 10. Internal movement equilibrium.

TABLE 5. Data extract ($|k_c v_c| \approx |k_r v_r|$).

Log	Id	N.Id	Distance	Cohesion	Repulsion
7	76	91	55.39031	276.95156	276.94684
24	75	6	55.39032	276.95160	276.94661
32	72	38	55.39002	276.95013	276.95370
35	63	64	55.39022	276.95113	276.94887

magnitude of the resultant vector is zero. Table 5 is an extract of data from the simulator. The data shows near equilibrium, but due to the dynamic nature of a swarm system, no agents meet the condition fully.

V. INTER-AGENT DISTANCE METRIC

The inter-agent distance metric considers a swarm in terms of how the agents are physically distributed: i.e. only the inter-agent distances and their standard deviation are considered. The standard deviation indicates the extent to which the swarm is out of balance and therefore the degree of movement required to rebalance the swarm structure. If the standard deviation is zero then all the agents are evenly spaced.

Navarro and Fernando describe a *mean distance error* metric that is based on the variations in distances between agents [22]. This is the same as the standard deviation of the distance-based internal movement metric described in this section.

Equation 8 calculates the mean distance of an agent b to its neighbors $nbr(b)$.

$$\mu_d(b) = \frac{1}{|nbr(b)|} \left(\sum_{b' \in nbr(b)} |bb'| \right) \quad (8)$$

The mean distance for a swarm is given by Equation 9. All the inter-agent distances are included for the swarm (S).

$$\mu_d(S) = \frac{\sum_{b \in S} \sum_{b' \in nbr(b)} |bb'|}{\sum_{b \in S} |nbr(b)|} \quad (9)$$

The mean distance provides an indication of the large scale structure of the swarm. However it is not sufficient to give an indication of the internal distribution of the agents.

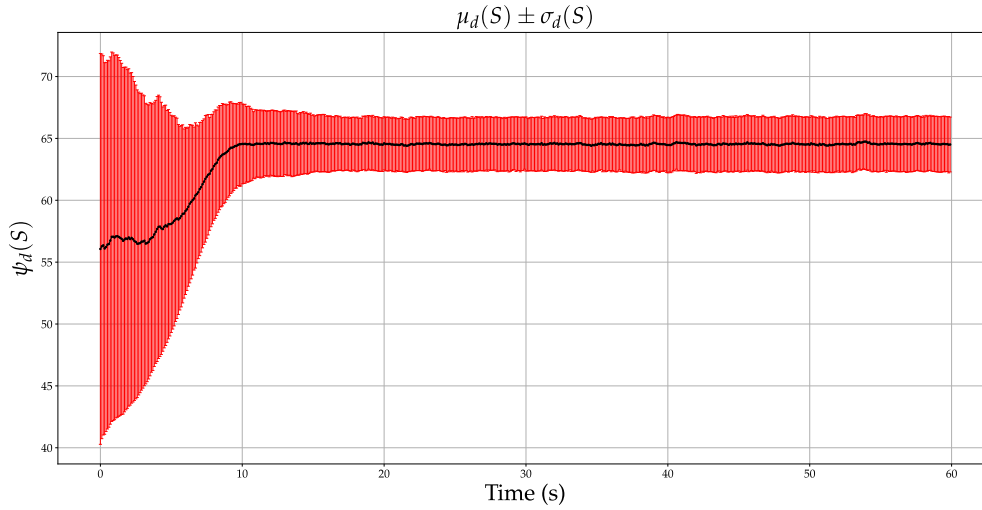


FIGURE 11. Baseline internal movement - distance.

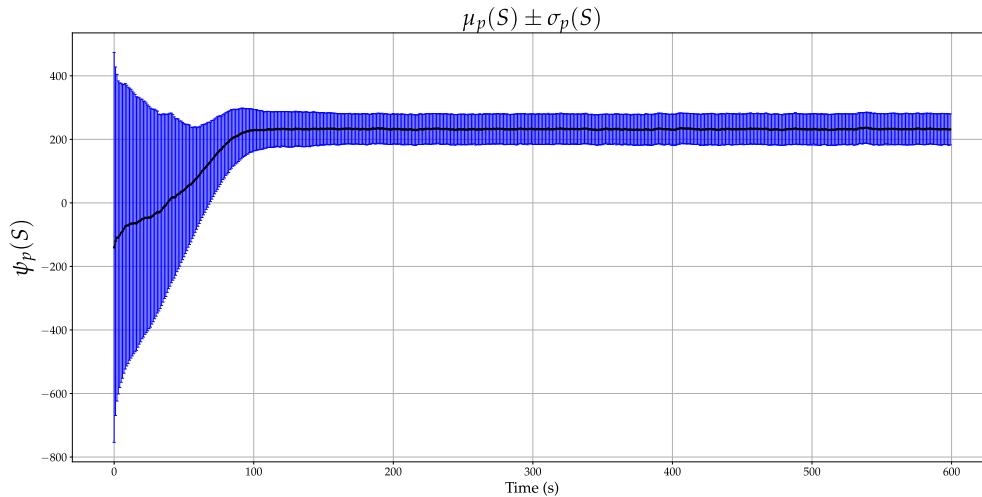


FIGURE 12. Baseline internal movement - cohesion/repulsion.

The standard deviation clarifies the distribution within the swarm as shown in Equation 10.

$$\sigma_d(S) = \sqrt{\frac{\sum_{b \in S} \sum_{b' \in nbr(b)} (|bb'| - \mu_d(S))^2}{\sum_{b \in S} |nbr(b)|}} \quad (10)$$

The distance-based metric for the internal distribution of the agents is therefore $\mu_d(S)$ and $\sigma_d(S)$. This can be written informally as:

$$\psi_d(S) = \mu_d(S) \pm \sigma_d(S) \quad (11)$$

Figure 11 shows a short example of the evolution of the metric over time.

VI. COHESION/REPULSION METRIC

This paper defines the *cohesion/repulsion metric* for an agent as the length of $v(b)$, as defined in Equation 5: that is the

size of the weighted effects of cohesion and repulsion caused by neighbors of the agent. This number carries a negative sign if repulsion dominates and a positive sign if cohesion dominates. This is summarized in Equation 12.

If the overall cohesion/repulsion metric of the swarm is negative, the swarm's tendency is to expand, as the repulsion is greater than cohesion. This is seen in the disorganized stage of a swarm (Fig. 12). The disorganized stage is the initial random deployment of the agents. If the overall cohesion/repulsion metric is positive, the swarm's tendency is to contract, and suggests that the swarm is a cohesive entity. This could also be described as the swarm being 'sticky', as the agents' tendency is to 'pull' towards each other.

The cohesion/repulsion metric on its own does not give a complete measure of a swarm's internal state. There needs to be a qualifying component to the metric that identifies the degree of deviation in the resultant magnitudes; this is the *jitter*. The smaller the deviation, the more uniform the

structure of the swarm. These two components identify the degree to which a swarm has progressed towards a stable state.

The cohesion/repulsion metric provides a view of the swarm's state through the balance between the repulsive and the cohesive vectors that are being applied to each agent. The deviation component identifies the degree to which the swarm has stabilized. The ideal status for inter-agent interactions would be for the agents to have a cohesion/repulsion metric of zero or above. This would indicate that the agents are distributed such that they are at their distribution limit (outer most range of the cohesion field) or at a level that causes the agents to 'pull' together. The ideal degree of deviation is zero as this indicates an even distribution of agents. These two aspects of a swarm's features are not considered by Gazi and Passino [14] or Barnes *et al.* [7] as a means of quantifying the structure of a swarm in terms of stability.

Section III discussed calculation of resultant cohesion (Equation 2) and resultant repulsion (Equation 3) vectors for each agent, deriving a combined cohesion/repulsion vector, as shown in Equation 5.

Now, we define for each agent b a *cohesion/repulsion scalar*, $P(b)$, to measure the influence of the neighbors of b as cohesive (positive) or repulsive (negative); shown in Equation 12.

$$P(b) = \begin{cases} |v(b)| & \text{if } |k_c v_c(b)| > |k_r v_r(b)| \\ -|v(b)| & \text{otherwise} \end{cases} \quad (12)$$

Although it is possible for $v(b)$ to be null there could still be variation in the constituent components. The variation calculation (standard deviation) is shown in Equation 14.

The mean cohesion/repulsion scalar for the swarm is now given by Equation 13.

$$\mu_p(S) = \frac{\sum_{b \in S} P(b)}{\sum_{b \in S} |nbr(b)| + \sum_{b \in S} |rep(b)|} \quad (13)$$

The standard deviation associated with this mean is calculated as Equation 14.

$$\sigma_p(S) = \sqrt{\frac{\sum_{b \in S} (P(b) - \mu_p(S))^2}{\sum_{b \in S} |nbr(b)| + \sum_{b \in S} |rep(b)|}} \quad (14)$$

The metric for the internal movement is this pair of numbers, the mean and standard deviation of the swarm's internal *cohesion/repulsion*. The pair $\mu_p(S)$, $\sigma_p(S)$ may be written informally as:

$$\psi_p(S) = \mu_p(S) \pm \sigma_p(S) \quad (15)$$

Figure 12 shows a short example of the evolution of the metric over time.

VII. APPLICATION: AREA FILLING

Area filling is a technique used to fill an enclosed area with agents such that they are distributed as 'effectively' as possible throughout the area.

Filling an area can be applied to tasks that require an unknown environment to be analyzed or surveyed. Consider a disaster area following a landslide or a building collapse following an earthquake; the movement in the land and buildings will produce areas that are unmapped. The unmapped areas may require investigation to locate people, or resources, or to create some form of sensor network to analyse the conditions within the area, such as creating a heat map or identifying the location of toxic gases.

The concept of using a swarm to provide coverage over unknown areas is a current area of research. Alvissalim *et al.* [2] discuss the application of commercially available drones to provide a communication infrastructure across an unmapped disaster area. Scheutz and Bauer [25] (2006) use both cohesion and repulsion as a mechanism to create coverage of an area that requires protection in an adversarial environment.

The concept behind the area filling is to increase both the repulsion and cohesion fields over a period of time. This increase in the field effects makes the agents increase the distance between each other, expanding the swarm as a whole. The expansion increases until, due to boundary compression, the swarm is unable to expand further. In an extreme case the expansion will result in a set of field effects that create a mesh-based swarm structure rather than the desired hexagonal structure that the field effects should create. Identifying the *cohesion/repulsion metric* falling below zero indicates area saturation has occurred.

To test this hypothesis a swarm is modelled in the simulator. The model consists of an obstacle-based enclosed space and a swarm consisting of 60 agents. The experimental parameters for the simulation are shown in Table 6

The field effects are incremented in turn, cohesion range followed by repulsion range. After each pair of increments,

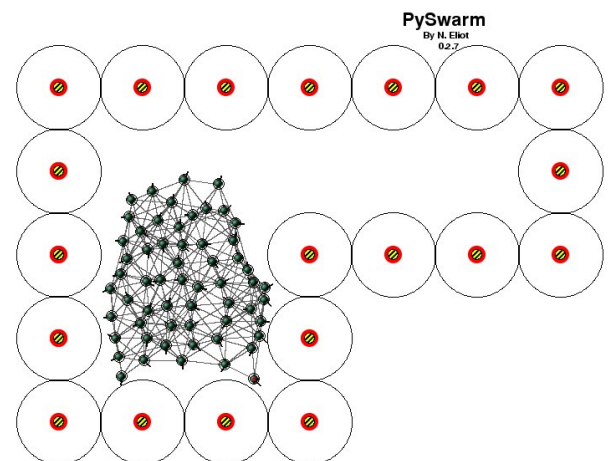


FIGURE 13. Flood fill Start.

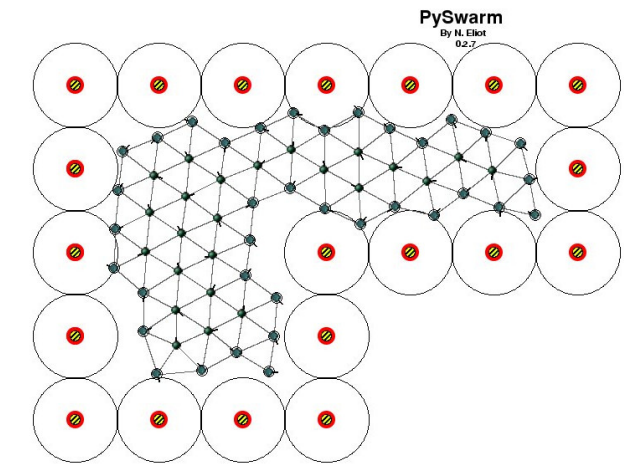


FIGURE 14. Data Capture Start.

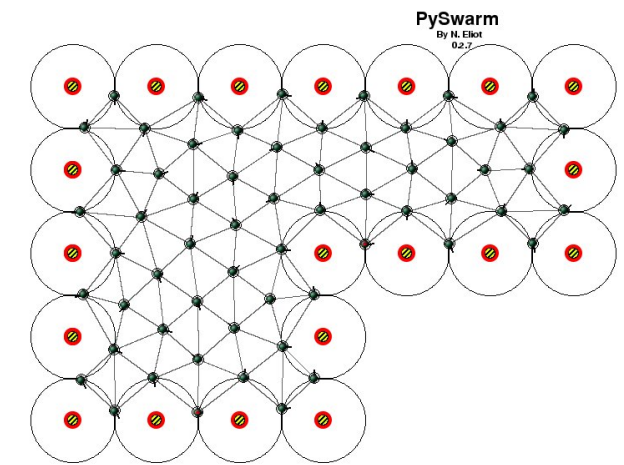


FIGURE 15. Flood fill End.

the swarm is allowed to redistribute itself and stabilize. Table 7 shows the field effect settings that are used for the simulation. The field effects are selected so as to ensure the

TABLE 6. Swarm parameters.

Parameter	Value	Description
k_c	5	weight adjuster for cohesion bias
k_r	60	weight adjuster for repulsion bias
Sample rate	100ms	proximity sensor rate
Speed	20 units/s	agent speed

TABLE 7. Field effect expansion sequence.

Weight component	1	2	3	4	5	6	7	
Repulsion Boundary	50	-	60	-	70	-	80	units
Neighbour Distance	60	70	-	80	-	90	-	units

swarm parameters have the potential to create a hexagonal swarm. Figure 13 shows the initial swarm deployment.

Figure 14 shows the point at which data logging starts. The swarm at this stage has expanded into the space but is not filling it. This presents itself as a positive cohesion/repulsion metric value, which indicates the swarm is cohesive and the agents are still “pulling” together. Fig. 15 shows the final swarm distribution for the simulation. At this point the cohesion/repulsion metric has fallen below zero, indicating the swarm is trying to expand.

Figure 16 shows the cohesion/repulsion metric for the swarm for the entire simulation. Between 100 seconds and 105 seconds there is a significant change in the magnitude where the value becomes negative. This indicates that there is a compression effect on the swarm and, hence, that the area has been filled.

Figure 17 shows the distance metric for the same simulation. The initial deployment is shown at 0 seconds, followed by incremental changes in the distances, and variance of the distances, as each increment is made in the field effects. The simulation terminates after 120 seconds. There is no feature of this graph which indicates that the area is filled. The cohesion/repulsion metric clearly tells us more.

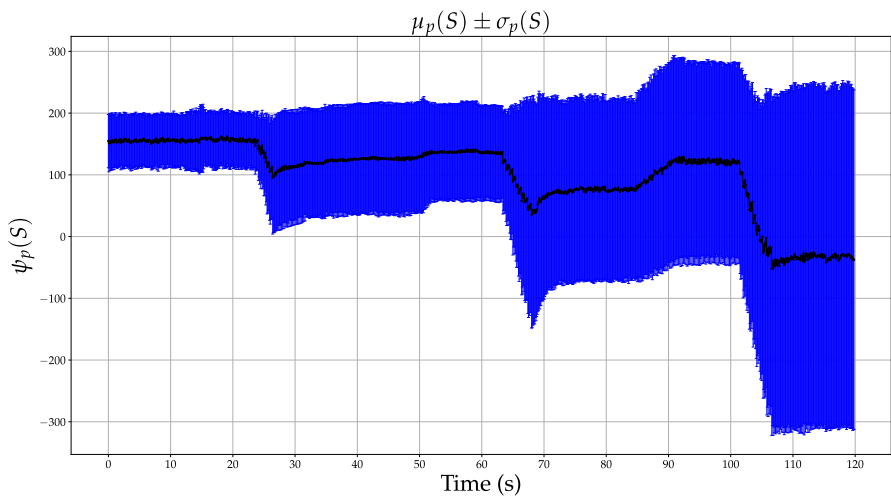


FIGURE 16. Cohesion/Repulsion metric 0-120 seconds.

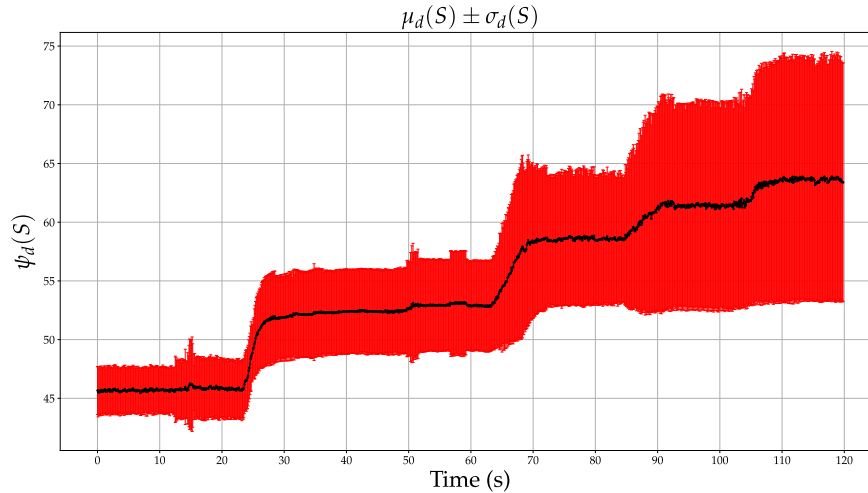


FIGURE 17. Distance metric 0-120 seconds.

VIII. CONCLUSION

Distance-based metrics measure a swarm's structure in terms of the position of agents. This is independent of the algorithm that produces the swarm's structure. It provides an analysis of agent distribution at a point in time but does not allow the identification of the future behavior of a swarm as 'expanding', 'stable' or 'contracting'.

The new *cohesion/repulsion* metric is based upon the algorithms that have produced the structure and is therefore an analysis of how the algorithms are performing rather than the structure presented. The new metric also provides an indication of a swarm's potential future movement. This is independent of the physical distribution of the agents within the swarm. The metric also provides the same information as the distance metric in terms of agent distribution when the field effects are consistent across all agents i.e. monolithic field effects. This is shown in Fig. 16 and 17, where the profiles of the two graphs are the same. In addition, the new metric provides a mechanism for evaluating swarms that use different algorithms to create a swarming effect and allows the algorithms' inter-agent interactions to be compared rather than the resultant structures that the algorithms create.

The two metrics can be used together to provide a deeper evaluation of a swarm's structure and 'potential'. Consider the following scenario: the repulsion field is increased but the internal distances do not change; as a result the *cohesion/repulsion metric* falls below zero. This indicates 'something' is confining the swarm's distribution as discussed in Section VII. This analysis could be used in identifying effective swarm distribution for the coverage of a sensor array as discussed by Ramaithitima *et al.* [23]

IX. FUTURE WORK

The new metric can be applied to the analysis of the effects of coordination algorithms upon a swarm's structure. An application to area filling has been discussed (Section VII). The metric can also be applied to identifying the effects of 'healing' algorithms, e.g. to remove 'voids' from a swarm.

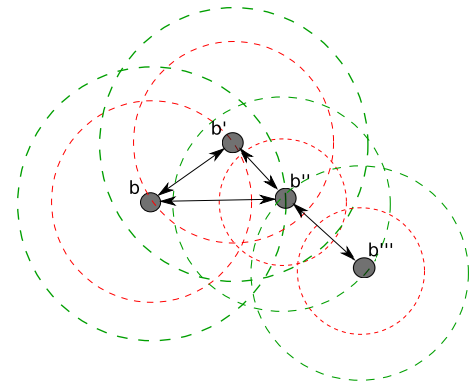


FIGURE 18. Heterogeneous field effects.

We plan to investigate this, and other, practical applications of the metric.

The detection of a swarm's perimeter [18]–[20], [28] identifies a subset of a swarm which can be used for coordination. Perimeter detection is widely used in swarm coordination [1], [3], [4], [29]. A possible adaptation of this area of research is to investigate the reduction of the field effects (cohesion and repulsion) on the perimeter so as to increase the agent density in this region of the swarm. This change in the swarm's structure could improve the granularity of analysis, e.g. when using sensor arrays at the frontier of the swarm. However, this approach leads to swarms of agents having heterogeneous field effects, as shown in Fig. 18. Clearly, the distance-based metric is not useful in the analysis of such swarms. We believe that the cohesion/repulsion metric introduced in this paper can identify a swarm's structure even when the swarm is heterogeneous, since the metric is based directly upon inter-agent field effects rather than the physical distribution of agents. This will also be the subject of forthcoming work.

REFERENCES

- [1] N. Agmon, S. Kraus, and G. A. Kaminka, "Multi-robot perimeter patrol in adversarial settings," in *Proc. IEEE Int. Conf. Robot. Automat. (ICRA)*, May 2008, pp. 2339–2345.

- [2] M. S. Alvisalim et al., "Swarm quadrotor robots for telecommunication network coverage area expansion in disaster area," in *Proc. SICE Annu. Conf. (SICE)*, Aug. 2012, pp. 2256–2261.
- [3] P. Andreou, D. Zeinalipour-Yazti, M. Andreou, P. K. Chrysanthis, and G. Samaras, "Perimeter-based data replication in mobile sensor networks," in *Proc. 10th Int. Conf. Mobile Data Manage., Syst., Services Middleware (MDM)*, May 2009, pp. 244–251.
- [4] P. Andreou, D. Zeinalipour-Yazti, P. K. Chrysanthis, and G. Samaras, "In-network data acquisition and replication in mobile sensor networks," *Distrib. Parallel Databases*, vol. 29, nos. 1–2, pp. 87–112, 2011.
- [5] L. Barnes, W. Alvis, M. Fields, K. Valavanis, and W. Moreno, "Heterogeneous swarm formation control using bivariate normal functions to generate potential fields," in *Proc. IEEE Workshop Distrib. Intell. Syst., Collective Intell. Appl. (DIS)*, Jun. 2006, pp. 85–94.
- [6] L. Barnes, W. Alvis, M. Fields, K. Valavanis, and W. Moreno, "Swarm formation control with potential fields formed by bivariate normal functions," in *Proc. 14th Medit. Conf. Control Autom. (MED)*, Jun. 2006, pp. 1–7.
- [7] L. Barnes, M. Fields, and K. Valavanis, "Unmanned ground vehicle swarm formation control using potential fields," in *Proc. Medit. Conf. Control Autom. (MED)*, Jun. 2007, pp. 1–8.
- [8] D. J. Bennet and C. R. McInnes, "Verifiable control of a swarm of unmanned aerial vehicles," *J. Aerosp. Eng.*, vol. 223, no. 7, pp. 939–953, 2009.
- [9] S. W. Ekanayake and P. N. Pathirana, "Formations of robotic swarm: An artificial force based approach," *Int. J. Adv. Robotic Syst.*, vol. 7, no. 3, pp. 173–190, 2010.
- [10] V. Gazi and K. M. Passino, "Stability analysis of social foraging swarms," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 34, no. 1, pp. 539–557, Feb. 2004.
- [11] V. Gazi and K. M. Passino, "Stability analysis of swarms in an environment with an attractant/repellent profile," in *Proc. Amer. Control Conf.*, vol. 3, May 2002, pp. 1819–1824.
- [12] V. Gazi and K. M. Passino, "A class of attractions/repulsion functions for stable swarm aggregations," *Int. J. Control*, vol. 77, no. 18, pp. 1567–1579, 2004.
- [13] V. Gazi and K. M. Passino, "Stability of a one-dimensional discrete-time asynchronous swarm," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 35, no. 4, pp. 834–841, Aug. 2005.
- [14] V. Gazi and K. M. Passino, *Swarm Stability and Optimization*. Berlin, Germany: Springer-Verlag, 2011.
- [15] P. Gurfil and E. Kivelevitch, "Flock properties effect on task assignment and formation flying of cooperating unmanned aerial vehicles," *Proc. Inst. Mech. Eng. G, J. Aerosp. Eng.*, vol. 221, no. 3, pp. 401–416, 2007.
- [16] J. Hereford, "Analysis of BEECLUST swarm algorithm," in *Proc. IEEE Symp. Swarm Intell. (SIS)*, Apr. 2011, pp. 1–7.
- [17] S. P. Hou, C. C. Cheah, and J. J. E. Slotine, "Dynamic region following formation control for a swarm of robots," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2009, pp. 1929–1934.
- [18] J. D. McLurkin, IV, "Analysis and implementation of distributed algorithms for multi-robot systems," Ph.D. dissertation, Dept. Elect. Eng. Comput. Sci., Massachusetts Inst. Technol., Cambridge, MA, USA, 2008.
- [19] S.-Y. Jung and M. A. Goodrich, "Multi-robot perimeter-shaping through mediator-based swarm control," in *Proc. 16th Int. Conf. Adv. Robot. (ICAR)*, Nov. 2013, pp. 1–6.
- [20] J. McLurkin and E. D. Demaine, "A distributed boundary detection algorithm for multi-robot systems," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2009, pp. 4791–4798.
- [21] Y. Mohan and S. G. Ponnambalam, "An extensive review of research in swarm robotics," in *Proc. World Congr. Nature Biologically Inspired Comput. (NaBIC)*, Dec. 2009, pp. 140–145.
- [22] I. Navarro and F. Matía, "A proposal of a set of metrics for collective movement of robots," in *Proc. Workshop Good Exp. Methodol. Robot.*, 2009, pp. 1–6.
- [23] R. Ramaithitima, M. Whitzer, S. Bhattacharya, and V. Kumar, "Sensor coverage robot swarms using local sensing without metric information," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2015, pp. 3408–3415.
- [24] C. W. Reynolds, "Flocks, herds and schools: A distributed behavioral model," *ACM SIGGRAPH Comput. Graph.*, vol. 21, no. 4, pp. 25–34, Jul. 1987.
- [25] M. Scheutz and P. Bauer, "A scalable, robust, ultra-low complexity agent swarm for area coverage and interception tasks," in *Proc. IEEE Conf. Comput. Aided Control Syst. Design, IEEE Int. Conf. Control Appl., IEEE Int. Symp. Intell. Control*, Oct. 2006, pp. 1258–1263.
- [26] F. E. Schneider and D. Wildermuth, "A potential field based approach to multi robot formation navigation," in *Proc. IEEE Int. Conf. Robot., Intell. Syst. Signal Process.*, vol. 1, Oct. 2003, pp. 680–685.
- [27] Y. Wu, Y. Chen, and L. Wang, "The evaluation of emergent behaviors on an improved Vicsek model," in *Proc. Int. Conf. Soft Comput. Pattern Recognit. (SoCPaR)*, Oct. 2011, pp. 173–178.
- [28] D. Zeinalipour-Yazti, P. Andreou, P. K. Chrysanthis, and G. Samaras, "SenseSwarm: A perimeter-based data acquisition framework for mobile sensor networks," in *Proc. 4th Workshop Data Manage. Sensor Netw., Conjunct 33rd Int. Conf. Very Large Data Bases (DMSN)*, New York, NY, USA, 2007, pp. 13–18.
- [29] G. Zhang, G. K. Fricke, and D. P. Garg, "Spill detection and perimeter surveillance via distributed swarming agents," *IEEE/ASME Trans. Mechatronics*, vol. 18, no. 1, pp. 121–129, Feb. 2013.



NEIL ELIOT received the B.Sc., Pg.C., and Ph.D. degrees in computing from Northumbria University in 1994, 1989, and 2017, respectively. He was with the chemical industry and the NHS. He has been a Senior Lecturer with Northumbria University since 1998. His research interests include swarm theory and cybersecurity.



DAVID KENDALL received the M.A. degree in literae humaniores from the New College, Oxford University, and the M.Sc. and Ph.D. degrees in computing science from Newcastle University. He was a Research Associate with Newcastle University from 1987 to 1989, following experience as a Senior Software Engineer in industry from 1983 to 1986. He has been a Senior Lecturer in computing with Northumbria University since 1989, except for a brief period as a Lecturer in computer science at Durham University from 2001 to 2002. His research interests include formal methods, embedded systems, distributed systems, security, and privacy.



MICHAEL BROCKWAY received the bachelor's degree in pure and applied mathematics, the master's degree in mathematics (mathematical logic and category theory), and the Ph.D. degree in computer science (formal methods for distributed embedded systems) from Northumbria University in 1974, 1975, and 2010, respectively. Prior to his Ph.D. degree, he was with the teaching and computing industry. He has been a Senior Lecturer with Northumbria University for 17 years.

...